OPEN ACCESS Freely available online

PLOS ONE

Using the Fast Fourier Transform to Accelerate the Computational Search for RNA Conformational Switches

Evan Senter[1], Saad Sheikh[2], Ivan Dotu[1], Yann Ponty[3], Peter Clote[1]*

1 Biology Department, Boston College, Chestnut Hill, Massachusetts, United States of America, **2** Computer Science Department, University of Florida, Gainesville, Florida, United States of America, **3** Laboratoire d'Informatique, Ecole Polytechnique, Palaiseau, France

Abstract

Using complex roots of unity and the Fast Fourier Transform, we design a new thermodynamics-based algorithm, FFTbor, that computes the Boltzmann probability that secondary structures differ by k base pairs from an arbitrary initial structure of a given RNA sequence. The algorithm, which runs in quartic time $O(n^4)$ and quadratic space $O(n^2)$, is used to determine the correlation between kinetic folding speed and the *ruggedness* of the energy landscape, and to predict the location of riboswitch expression platform candidates. A web server is available at http://bioinformatics.bc.edu/clotelab/FFTbor/.

Citation: Senter E, Sheikh S, Dotu I, Ponty Y, Clote P (2012) Using the Fast Fourier Transform to Accelerate the Computational Search for RNA Conformational Switches. PLoS ONE 7(12): e50506. doi:10.1371/journal.pone.0050506

Editor: Freddie Salsbury Jr, Wake Forest University, United States of America

Received September 21, 2012; **Accepted** October 26, 2012; **Published** December 19, 2012

Funding: Source of funding provided by National Science Foundation grants DMS-1016618 and DMS-0817971 to PC, www.nsf.gov. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: clote@bc.edu

Introduction

In [1], we developed a dynamic programming algorithm, RNAbor, pronounced *RNA neighbor*, which simultaneously computes for each integer k, the Boltzmann probability $p_k = \frac{Z_k}{Z}$ of the subensemble of structures whose base pair distance to a given *initial*, or *reference*, structure S^* is k. (Here, Z denotes the partition function, defined as the sum of all Boltzmann factors $\exp(-E(S)/RT)$, over all secondary structures S of a given RNA sequence, and R denotes the universal gas constant and T absolute temperature. Similarly Z_k denotes the sum of all Boltzmann factors of all structures S, whose base pair distance to the initial structure S^* is exactly k.) RNAbor stores the value of the (partial) partition functions $Z_k(i,j)$ for all $1 \le i \le j \le n$ and $0 \le k \le n$, each of which requires quadratic time to compute. Thus it follows that RNAbor runs in time $O(n^5)$ and space $O(n^3)$, which severely limits its applicability to genomic annotation. This restriction is somewhat mitigated by the fact that in [2], we showed how to use sampling [3] to efficiently approximate RNAbor in cubic time $O(n^3)$ and quadratic space $O(n^2)$, *provided* that the starting structure S^* is the minimum free energy (MFE) structure. We expect that a more efficient version of RNAbor could be used in applications in genomics and synthetic biology, to detect potential conformational switches – RNA sequences containing two or more (distinct) metastable structures.

In this paper, we describe a radically different algorithm, FFTbor, prounounced *FFT neighbor*, that uses polynomial interpolation to compute the coefficients $p_0, \ldots, p_{n-1}$ of the polynomial

$$p(x) = p_0 + p_1 x + p_2 x^2 + \cdots + p_{n-1} x^{n-1}, \qquad (1)$$

where p_k is defined by $p_k = \frac{Z_k}{Z}$. Due to severe numerical instability issues in both the Lagrange interpolation formula and in Gaussian elimination, we employ the Fast Fourier Transform (FFT) to compute the inverse Discrete Fourier Transform (DFT) on values $y_0, \ldots, y_{n-1}$, where $y_k = p(\omega^k)$ and $\omega = e^{2\pi i/n}$ is the principal nth complex root of unity and $p(x)$ is defined in (1). This gives rise to an improved version of RNAbor, denoted FFTbor, which runs in time $O(n^4)$ and space $O(n^2)$. Once two metastable structures S_1, S_2 are identified, we can subsequently evaluate the feasibility of transition between structures S_1 and S_2, by computing the *barrier energy* using algorithms, such as that described in Dotu et al. [4] or Flamm et al. [5].

Background

Let $\mathbf{s} = s_1, \ldots, s_n$ denote an RNA sequence, i.e. a sequence of letters in the alphabet of nucleotides $\{A,C,G,U\}$. A secondary structure S is a set of base pairs (i,j), where $1 \le i \le i+\theta < j \le n$ and $\theta \ge 0$ represents the minimum number of unpaired nucleotides in a hairpin loop (due to steric constraints, θ is usually taken to be 3), such that if (i,j) and (x,y) both belong to S, then $i = x \Leftarrow j = y$ (a nucleotide is involved in at most one base pair) and $i < x < j \Leftarrow i < y < j$ (no pseudoknots are allowed).

The secondary structure S is *compatible* with s if for every base pair (i,j) in S, the pair (s_i, s_j) is contained in the set $\mathbb{B} = \{(A,U),(U,A),(G,C),(C,G),(G,U),(U,G)\}$ of six Watson-Crick and wobble base pairs. Often we write that S is a secondary structure *on* $\mathbf{s}$, or equivalently, a secondary structure *of* $\mathbf{s}$, in place of stating that S is compatible with $\mathbf{s}$. Throughout this paper, by *structure*, we always mean a secondary structure which is compatible with an arbitrary, but fixed RNA sequence $\mathbf{s} = s_1, \ldots, s_n$.

Given two secondary structures S, T on $\mathbf{s}$, we define the base pair distance d_{BP} between S and T to be the number of base pairs that they have that are not in common, i.e.

$$d_{BP}(S,T)=|S\cup T|-|S\cap T|. \quad (2)$$

Structures S,T are said to be k- *neighbors* if $d_{BP}(S,T)=k$.

For $1\le i\le j\le n$, let $S_{[i,j]}$ denote the restriction of S to interval $[i,j]$ of $\mathbf{s}$, i.e. the set of base pairs $S_{[i,j]}=\{(x,y) : i\le x<y\le j,(x,y)\in S\}$. The notion of k-neighbor can also be applied to restrictions of secondary structures; i.e. a secondary structure $T_{[i,j]}$ is a *k-neighbor* of $S_{[i,j]}$ if

$$d_{BP}(S_{[i,j]},T_{[i,j]})=$$
$$|\{(x,y) : i\le x<y\le j,(x,y)\in S-T \text{ or } (x,y)\in T-S\}|=k.$$

In the following, we often omit the sequence s and initial secondary structure S^* in our notation, since these are arbitrary, but fixed. In particular, we write $Z^k_{i,j}=Z^k_{i,j}(\mathbf{s},S^*)$ – see following paragraph for definitions.

Given an RNA sequence $\mathbf{s}=s_1,\ldots,s_n$ and compatible secondary structure S^*, let Z^k denote the sum of the Boltzmann factors $\exp(-E(S)/RT)$ of all k-neighbors S of S^*; i.e.

$$Z^k=Z^k_{1,n}=\sum_{\substack{S \text{ such that}\\ d_{BP}(S,S^*)=k}} e^{\frac{-E(S)}{RT}}$$

where $E(S)$ denotes the Turner (nearest neighbor) energy [6,7] of S, $R=0.00198$ kcal/mol denotes the universal gas constant and T denotes absolute temperature. Since the maximum base pair distance between a given initial structure S^* and any other structure S on RNA sequence $\mathbf{s}=s_1,\ldots,s_n$ must satisfy

$$d_{BP}(S,S^*)\le|S^*|+\lfloor\frac{n-\theta}{2}\rfloor\le n \quad (3)$$

it follows that the full partition function

$$Z=Z_{1,n}=\sum_{k=0}^{n}Z^k_{1,n}. \quad (4)$$

Moreover, since $\theta=3$, we need to compute at most the values $Z^0_{1,n},\ldots,Z^{n-1}_{1,n}$ – this observation will later prove useful. The Boltzmann probability $P[d_{BP}(S,S^*)=k]$ that a secondary structure S has base pair distance k from the initial structure S^* can be defined from the partition function by

$$p(k) := \frac{Z^k_{1,n}}{Z_{1,n}}.$$

By graphing the probabilities p_k as a function of k, we expect to see one or more peaks at base pair distance k when there is a meta-stable (low energy) structure S at base pair distance k from S^*. See Figure 1 for an illustration.

Recursions for structural neighbors

For the rest of the paper, we consider both $\mathbf{s}$ as well as the secondary structure S^* on $\mathbf{s}$ to be fixed. We now recall the recursions from Freyhult et al. [8] to determine the partition function $Z^k_{i,j}$ with respect to the Nussinov-Jacobson energy E_0 model [9], defined by -1 times the number of base pairs; i.e. $E_0(S)=-1\cdot|S|$. Although we describe here the recursions for the Nussinov-Jacobson model, for the sake of simplicity of exposition, both RNAbor [8] as well as our current software FFTbor, concern the Turner energy model, consisting of free energy parameters for stacked bases, hairpins, bulges, internal loops and multiloops. The full recursions for FFTbor are described for the the Turner energy model in the appendix.

The base case for $Z^k_{i,j}$ is given by

$$Z^0_{i,j}=1, \text{ for } i\le j, \quad (5)$$

since the only 0-neighbor to a structure S^* is the structure S^* itself, and

$$Z^k_{i,j}=0, \text{for } k>0,\ i\le j\le i+\theta, \quad (6)$$

since the empty structure is the only possible structure for a sequence shorter than $\theta+2$ nucleotides, and so there are no k-neighbors for $k>0$. The recursion used to compute $Z^k_{i,j}$ for $k>0$ and $j>i+\theta$ is

$$Z^k_{i,j}=Z^{k-b_0}_{i,j-1}+\sum_{\substack{(s_r,s_j)\in\mathbb{B},\ w+w'=k-b(r)\\ i\le r<j}}\sum \exp(-E_0(r,j)/RT)\cdot Z^w_{i,r-1}Z^{w'}_{r+1,j-1}, \quad (7)$$

where $E_0(r,j)=-1$ if positions r,j can pair in sequence s, and otherwise $E_0(r,j)=+\infty$. Additionally, $b_0=1$ if j is base-paired in $S^*_{[i,j]}$ and 0 otherwise, and $b(r)=d_{BP}(S^*_{[i,j]},S^*_{[i,r-1]}\cup S^*_{[r+1,j-1]}\cup\{(r,j)\})$. This holds since in a secondary structure $T_{[i,j]}$ on $s_i,\ldots,s_j$ that is a k-neighbor of $S^*_{[i,j]}$, either nucleotide j is unpaired in $[i,j]$ or it is paired to a nucleotide r such that $i\le r<j$. In this latter case it is enough to study the smaller sequence segments $[i,r-1]$ and $[r+1,j-1]$ noting that, except for (r,j), base pairs outside of these regions are not allowed, since there are no pseudoknots. In addition, for $d_{BP}(S^*_{[i,j]},T_{[i,j]})=k$ to hold, it is necessary for $w+w'=k-b(r)$ to hold, where $w=d_{BP}(S^*_{[i,r-1]},T_{[i,r-1]})$ and $w'=d_{BP}(S^*_{[r+1,j-1]},T_{[r+1,j-1]})$, since $b(r)$ is the number of base pairs that differ between $S^*_{[i,j]}$ and a structure $T_{[i,j]}$, due to the introduction of the base pair (r,j).

Methods

Given RNA sequence $\mathbf{s}$ and compatible initial structure S^*, we define the *polynomial*

$$\mathcal{Z}(x)=\sum_{k=0}^{n}c_kx^k \quad (8)$$

where coefficients $c_k=Z^k_{1,n}$. Moreover, because of (3) and the fact that the minimum number of unpaired bases in a hairpin loop θ is 3, we know that $c_n=0$, so that $\mathcal{Z}(x)$ is a polynomial of degree strictly less than n. If we evaluate the polynomial $\mathcal{Z}(x)$ for n distinct values

$$\mathcal{Z}(a_1)=y_1,\ldots,\mathcal{Z}(a_n)=y_n, \quad (9)$$

then the Lagrange polynomial interpolation formula guarantees that $\mathcal{Z}(x)=\sum_{k=1}^{n}y_kP_k(x)$, where the polynomials $P_k(x)$ have degree at most $n-1$ and are given by the Lagrange formula

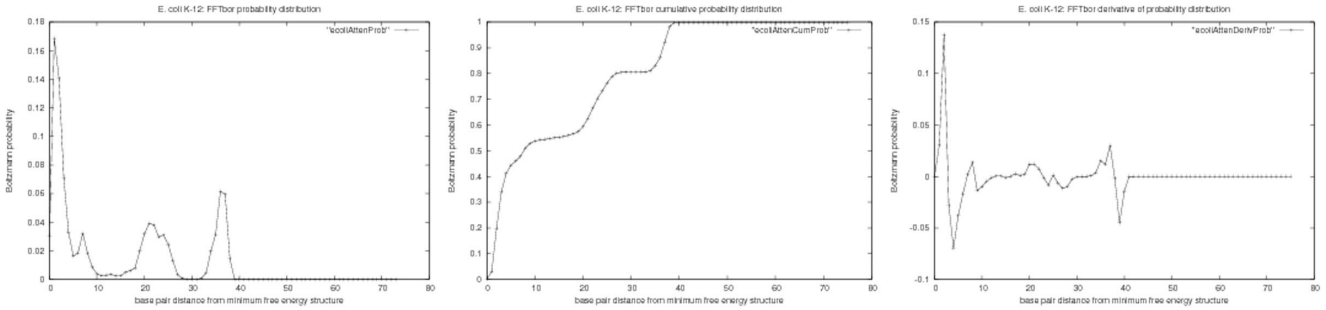

Figure 1. FFTbor output for the RNA attenuator for the phenylalanyl-tRNA synthetase (pheST) operon in *E. coli* K-12 substr. DH10B, located adjacent to the phenylalanyl-tRNA synthetase operon leader, with GenBank accession code CP000948.1/1887748-1887820 (complement). The x-axis represents base pair distance to the minimum free energy structure S^*; y-axis represents Boltzmann probability $p(k)$ that a structure has base pair distance k to S^*. *(Left)* Probability $P(d_{BP}(S,S^*)=k)$ that base pair distance to MFE structure is k. *(Center)* Cumulative probability $P(d_{BP}(S,S^*)\leq k)$ that base pair distance to MFE structure is at most k. *(Right)* Finite difference (Derivative) $P(k\leq d_{BP}(S,S^*)=k+1)$ of probability that base pair distance to MFE structure is k.
doi:10.1371/journal.pone.0050506.g001

$$P_k(x)=\frac{\prod_{i\neq k}(x-x_i)}{\prod_{i\neq k}(x_k-x_i)}. \tag{10}$$

Since the polynomials $P_k(x)$ can be explicitly computed, it follows that we can compute the coefficients c_k of polynomial $\mathcal{Z}(x)$. As we describe below, the evaluation of $\mathcal{Z}(x)$ for a fixed value of x can be done in time $O(n^3)$ and space $O(n^2)$. It follows that the coefficients $c_k=Z^k_{1,n}$ can be computed after n evaluations of $\mathcal{Z}(x)$, where the space for each evaluation of $\mathcal{Z}(x)$ is re-used; hence these evaluations can be performed in time $O(n^4)$ and space $O(n^2)$. Finally, Lagrange interpolation is clearly computable in time $O(n^3)$. Although this approach is theoretically sound, there are severe numerical stability issues related to the interpolation method [10], the choice of values $a_1,\ldots,a_n$ in the interpolation, and floating point arithmetic (round-off error) related to the astronomically large values of the partition functions $Z^k_{1,n}$, for $0\leq k<n$. After many unsuccessful approaches including scaling (see File S1), we obtained excellent results by interpolating the polynomial $p(x)$, defined in equation (1), rather than the polynomial $\mathcal{Z}(x)$, defined in equation (9), and performing interpolation with the Fast Fourier Transform (FFT) [11] where $\alpha_0,\ldots,\alpha_{n-1}$ are chosen to be nth complex roots of unity, $\alpha_k=e^{\frac{2\pi k}{n}}$. One advantage of the FFT is that interpolation can be performed in $O(n\log n)$ time, rather than the cubic time required by using the Lagrange formula (10) or by Gaussian elimination. Fewer numerical operations implies increased numerical stability in our application. Details now follow.

Recursions to compute the polynomial $\mathcal{Z}_{i,j}(x)$

Given an initial secondary structure S^* of a given RNA sequence $\mathbf{s}$, our goal is to compute

$$Z^k_{1,n}=\sum_{\substack{S \text{ such that}\\ d_{BP}(S,S^*)=k}} e^{\frac{-E_0(S)}{RT}} \tag{11}$$

where S can be any structure compatible with $\mathbf{s}$. As previously mentioned, the recurrence relation for RNAbor with respect to the Nussinov energy model E_0 is

$$Z^k_{i,j}=Z^{k-b_0}_{i,j-1}+\sum_{\substack{s_r s_j\in\mathbb{B},\\ i\leq r<j}}\left(e^{\frac{-E_0(r,j)}{RT}}\sum_{w+w'=k-b(r)} Z^w_{i,r-1}Z^{w'}_{r+1,j-1}\right) \tag{12}$$

where $E_0(r,j)=-1$ if r and j can base-pair and otherwise $+\infty$, and $b_0=1$ if j is base paired in $S^*_{[i,j]}$ and 0 otherwise, and $b(r)=d_{BP}(S^*_{[i,j]},S^*_{[i,r-1]}\cup S^*_{[r+1,j-1]}\cup\{(r,j)\})$. The following theorem shows that an analogous recursion can be used to compute the *polynomial* $\mathcal{Z}_{i,j}(x)$ defined by

$$\mathcal{Z}_{i,j}(x)=\sum_{k=0}^{n} c_k(i,j)x^k \tag{13}$$

where

$$c_k(i,j)=Z^k_{i,j}=\sum_{\substack{S \text{ such that}\\ d_{BP}(S,S^*_{[i,j]})=k}} e^{\frac{-E_0(S)}{RT}}.$$

Here, in the summation, S runs over structures on $s_i,\ldots,s_j$, which are k-neighbors of the restriction $S^*_{[i,j]}$ of initial structure S^* to interval $[i,j]$, and $E_0(S)=-1\cdot|S|$ denotes the Nussinov-Jacobson energy of S.

THEOREM 1: Let $s_1,\ldots,s_n$ be a given RNA sequence. For any integers $1\leq i\leq j\leq n$, let

$$\mathcal{Z}_{i,j}(x)=\sum_{k=0}^{n} c_k x^k$$

where

$$c_k(i,j)=Z^k_{i,j}.$$

Then for $i\leq j\leq i+\theta$, $\mathcal{Z}_{i,j}(x)=1$ and for $j>i+\theta$ we have the recurrence relation

$$\mathcal{Z}_{i,j}(x) = \mathcal{Z}_{i,j-1}(x)\cdot x^{b_0} + \sum_{\substack{sr s_j \in \mathbb{B}, \\ i \le r < j}} \left(e^{\frac{-E_0(r,j)}{RT}} \cdot \mathcal{Z}_{i,r-1}(x) \cdot \mathcal{Z}_{r+1,j-1}(x) \cdot x^{b(r)} \right). \quad (14)$$

where $b_0 = 1$ if j is base-paired in $S^*_{[i,j]}$ and 0 otherwise, and $b(r) = d_{BP}(S^*_{[i,j]}, S^*_{[i,r-1]} \cup S^*_{[r+1,j-1]} \cup \{(r,j)\})$.

PROOF: First, some notation is necessary. Recall that if F is an arbitrary polynomial [resp. analytic] function, then $[x^k]F(x)$ denotes the coefficient of x^k [resp. the kth Taylor coefficient in the Taylor expansion of F] – for instance, in equation (1), $[x^k]p(x) = p_k$, and in equation (9), $[x^k]\mathcal{Z}(x) = c_k(i,j)$.

By definition, it is clear that $\mathcal{Z}_{i,j}(x) = 1$ if $i \le j \le i+\theta$, where we recall that $\theta = 3$ is the minimum number of unpaired bases in a hairpin loop. For $j > i+\theta$, we have

$$\begin{aligned}
[x^k]\mathcal{Z}_{i,j}(x) = c_k(i,j) &= Z^k_{i,j} \\
&= Z^{k-b_0}_{i,j-1} + \sum_{r=i}^{j-1} \sum_{k_0+k_1=k-b(r)} e^{\frac{-E_0(r,j)}{RT}} \cdot Z^{k_0}_{i,r-1} \cdot Z^{k_1}_{r+1,j-1} \\
&= [x^{k-b_0}]\mathcal{Z}_{i,j-1}(x) + \sum_{r=i}^{j-1} \sum_{k_0+k_1=k-b(r)} e^{\frac{-E_0(r,j)}{RT}} \\
&\cdot \left\{ [x^{k_0}]\mathcal{Z}_{i,r-1}(x) \right\} \cdot \left\{ [x^{k_1}]\mathcal{Z}_{r-1,j-1}(x) \right\} \\
&= [x^{k-b_0}]\mathcal{Z}_{i,j-1}(x) + \sum_{r=i}^{j-1} \sum_{k_0+k_1=k-b(r)} e^{\frac{-E_0(r,j)}{RT}} \\
&\cdot [x^{k_0+k_1}]\mathcal{Z}_{i,r-1}(x)\mathcal{Z}_{r-1,j-1}(x).
\end{aligned}$$

By induction, the proof of the theorem now follows. □

Notice that if one were to compute all terms of the polynomial $\mathcal{Z}_{1,n}(x)$ by explicitly performing polynomial multiplications, then the computation would require $O(n^5)$ time and $O(n^3)$ space. Instead of explicitly performing polynomial expansion in *variable* x, we instantiate x to a fixed complex number $\alpha \in \mathbb{C}$, and apply the following recursion for this instantiation:

$$\mathcal{Z}_{i,j}(\alpha) = \mathcal{Z}_{i,j-1}(\alpha)\cdot \alpha^{b_0} + \sum_{\substack{(s_r,s_j)\in\mathbb{B}, \\ i \le r < j}} \left(e^{\frac{-E_0(r,j)}{RT}} \cdot \mathcal{Z}_{i,r-1}(\alpha) \cdot \mathcal{Z}_{r+1,j-1}(\alpha) \cdot \alpha^{b(r)} \right). \quad (15)$$

In this fashion, we can compute $\mathcal{Z}(\alpha) = \mathcal{Z}_{1,n}(\alpha)$ in $O(n^3)$ time and $O(n^2)$ space. For n distinct complex values $\alpha_0, \ldots, \alpha_{n-1}$, we can compute and save only the values $\mathcal{Z}(\alpha_0), \ldots, \mathcal{Z}(\alpha_{n-1})$, each time re-using the $O(n^2)$ space for the next computation of $\mathcal{Z}(\alpha_k)$. It follows that the computation resources used to determine the (column) vector

$$\mathbf{Y} = (y_0, \ldots, y_{n-1})^T = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_{n-1} \end{pmatrix} \quad (16)$$

where $y_0 = \mathcal{Z}(\alpha_0), \ldots, y_{n-1} = \mathcal{Z}(\alpha_{n-1})$ is thus quartic time $O(n^4)$ and quadratic space $O(n^2)$.

Polynomial interpolation using the FFT

Let $\omega = e^{2\pi i/n}$ be the principal nth complex root of unity. Recall that the Vandermonde matrix V_n is defined to be the $n \times n$ matrix, whose i,j entry is ω^{ij}; i.e.

$$V_n = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \cdots & \omega^{n-1} \\ 1 & \omega^2 & \omega^4 & \cdots & \omega^{2(n-1)} \\ 1 & \omega^3 & \omega^6 & \cdots & \omega^{3(n-1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \omega^{n-1} & \omega^{2(n-1)} & \cdots & \omega^{(n-1)(n-1)} \end{pmatrix}$$

The Fast Fourier Transform (FFT) is defined to be the $O(n \log n)$ algorithm to compute the Discrete Fourier Transform (DFT), defined as the matrix product $\mathbf{Y} = V_n\mathbf{A}$:

$$\begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_{n-1} \end{pmatrix} = V_n \cdot \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_{n-1} \end{pmatrix}$$

On page 837 of [11], it is shown that the (i,j) entry of V_n^{-1} is $\frac{\omega^{-ji}}{n}$ and that

$$a_j = \frac{1}{n}\sum_{k=0}^{n-1} y_k \omega^{-kj} \quad (17)$$

for $j = 0, \ldots, n-1$.

Since we defined $\mathbf{Y}$ in (16) by $\mathbf{Y} = (y_0, \ldots, y_{n-1})^T$, where $y_0 = \mathcal{Z}(\alpha_0), \ldots, y_{n-1} = \mathcal{Z}(\alpha_{n-1})$ and $\alpha_k = \omega^k \exp(\frac{k\cdot 2\pi i}{n})$, it follows that the coefficients $c_k = \mathbf{Z}^k_{1,n}$ in the polynomial $\mathcal{Z}(x) = c_0 + c_1 x + \cdots + c_{n-1}x^{n-1}$ defined in (8) can be computed, at least in principle, by using the FFT. It turns out, however, that the values of $\mathbf{Z}^k_{1,n}$ are so astronomically large, that the ensuing numerical instability makes even this approach infeasible for values of n that exceed 56 (data not shown). Nevertheless, our approach can be modified as follows. Define $\mathbf{Y}$ by

$\mathbf{Y}=(y_1,\ldots,y_n)^T$, where $y_1=\frac{\mathcal{Z}(\alpha_1)}{Z},\ldots,y_n=\frac{\mathcal{Z}(\alpha_n)}{Z}$, and Z is the partition function defined in (4). Using the FFT to compute the inverse DFT, it follows from (17) that we can compute the probabilities $p_0,\ldots,p_{n-1}$ that are coefficients of the polynomial $p(x)=p_0+p_1x+\cdots+p_{n-1}x^{n-1}$ defined in equation (1). For genomics applications, we are only interested in the m most significant digits of each p_k, as described in the pseudocode below.

ALGORITHM for FFTbor

This pseudocode computes the m most significant digits of probabilities $p_k=\frac{\mathbf{Z}^k_{1,n}}{\mathbf{Z}}$.

INPUT: RNA sequence $\mathbf{s}=s_1,\ldots,s_n$, and initial secondary structure S^* of $\mathbf{s}$, and integer m.

OUTPUT: Probabilities $p_k=\mathbf{Z}^k_{1,n}/\mathbf{Z}$ to m significant digits for $k=0,\ldots,n-1$.

1. generate roots of unity ω^k for $k=0,\ldots,n-1$, where $\omega=\exp(\frac{2\pi i}{n})$ and $i=\sqrt{-1}$
2. note that the partition function $\mathbf{Z}=y_0=\mathcal{Z}(\omega^0)$
3. for $k=0$ to $n-1$
4. compute $y_k=\mathcal{Z}(\omega^k)$ using recursion (15)
5. $y_k=10^m\cdot\frac{y_k}{Z}$ //normalize y_k
6. compute $P=(p_0,\ldots,p_{n-1})^T$ where $p_j=\frac{\sum_{k=0}^{n-1}a_k\omega^{-kj}}{n}$ by using FFT in (17)
7. for $k=0$ to $n-1$
8. $p_k=\lfloor 10^m\cdot p_k\rfloor\cdot\frac{1}{10^m}$
9. //truncate to m most significant digits

Speed-up in our implementation of FFTbor. In this subsection, we show that we need only evaluate the polynomial $\mathcal{Z}(x)$, as defined in equation (8), for $n/2$ of the complex nth roots of unity. It is first necessary to recall the definition of complex conjugate. Recall that the complex conjugate of z is denoted by $\overline{z}$; i.e. if $z=a+bi$ where $a,b\in\mathbb{R}$ are real numbers and $i=\sqrt{-1}$, then $\overline{z}=a-bi$.

LEMMA 1: If $\mathcal{Z}(x)$ is the complex polynomial defined in equation (8), then for any complex nth root of unity α, it is the case that $\mathcal{Z}(\overline{\alpha})=\overline{\mathcal{Z}(\alpha)}$. In other words, if α is a complex nth root of unity of the form $a+bi$, where $a,b\in\mathbb{R}$ and $b>0$, and if $\mathcal{Z}(a+bi)=A+Bi$ where $A,B\in\mathbb{R}$, then it is the case that

$$\mathcal{Z}(a-bi)=A-Bi.$$

PROOF: Letting $i=\sqrt{-1}$, if $\theta=\frac{2\pi}{n}$, then $\omega=e^{i\theta}=\cos(\theta)+i\sin(\theta)$ is the principal nth complex root of unity, and $1=\omega^0,\ldots,e^{(n-1)\cdot i\theta}=\omega^{n-1}$ together constitute the complete collection of all nth complex roots of unity – i.e. the n unique solutions of of the equation $x^n-1=0$ over the field $\mathbb{C}$ of complex numbers. Clearly, for any $1\le r<n$, $e^{-ir\theta}=1\cdot e^{-ir\theta}=e^{2\pi i}\cdot e^{-ir\theta}=e^{i(2\pi-r\theta)}=e^{i(n\theta-r\theta)}=e^{i\theta(n-r)}$. Moreover, if $\omega^r=e^{ir\theta}=a+bi$ where $b>0$, then we have $e^{-ir\theta}=a-bi$. It follows that for any nth root of unity of the form $a+bi$, where $b>0$, the number $a-bi$ is also an nth root of unity.

Recall that $\mathcal{Z}(x)=\sum_{k=0}^n c_kx^k$, where $c_k\in\mathbb{R}$ are real numbers representing the partition function $Z^k_{1,n}$ over all secondary structures of a given RNA sequence $s_1,\ldots,s_n$, whose base pair distance from initial structure S^* is k. Thus, in order to prove the lemma, it suffices to show that for all values $k=0,\ldots,n-1$, if $a+bi$ is a complex nth root of unity, where $a,b\in\mathbb{R}$ and $b>0$, and if $(a+bi)^k=C+Di$ where $C,D\in\mathbb{R}$, *then* $(a-bi)^k=C-Di$. Indeed, we have the following.

$$(a+bi)^m=\sum_{k=0}^m\binom{m}{k}a^{m-k}\cdot(bi)^k$$

$$(bi)^k=\begin{cases}b^k & \text{if } k\equiv 0 \bmod 4\\ ib^k & \text{if } k\equiv 1 \bmod 4\\ -b^k & \text{if } k\equiv 2 \bmod 4\\ -ib^k & \text{if } k\equiv 3 \bmod 4\end{cases}$$

$$(a-bi)^m=\sum_{k=0}^m\binom{m}{k}a^{m-k}\cdot(-bi)^k$$

$$(-bi)^k=\begin{cases}b^k & \text{if } k\equiv 0 \bmod 4\\ -ib^k & \text{if } k\equiv 1 \bmod 4\\ -b^k & \text{if } k\equiv 2 \bmod 4\\ ib^k & \text{if } k\equiv 3 \bmod 4\end{cases}$$

It follows that each term of the form $a^{m-k}\cdot(bi)^k$, for $k=0,\ldots,m$, is the complex conjugate of $a^{m-k}\cdot(-bi)^k$, and thus $(a+bi)^m$ is the complex conjugate of $(a-bi)^m$. Since $\mathcal{Z}(a+bi)$ is a sum of terms of the form $c_k(a+bi)^k$, it follows that $\mathcal{Z}(a-bi)$ is the complex conjugate of $\mathcal{Z}(a+bi)$. This completes the proof of the lemma. □

Lemma 1 immediately entails that we need only evaluate $\mathcal{Z}(x)$ on $n/2$ many of the complex nth roots of unity – namely, those of the form $a+bi$, where $b\ge 0$. The remaining values of $\mathcal{Z}(x)$ are obtained by taking conplex conjugates of the first $n/2$ values. This, along with a precomputation of powers of the complex nth roots of unity, leads to an enormous performance speed-up in our implementation of FFTbor.

Results

Applications of FFTbor

In this section, we consider two applications of FFTbor: *(i)* correlation between kinetic folding speed and the *ruggedness* of the energy landscape near the minimum free energy structure, *(ii)* computational detection of riboswitch expression platform candidates.

Kinetic folding speed and energy landscape ruggedness. The output of FFTbor, as shown in Figure 2, is a probability distribution, where the x-axis represents the base pair distance from an arbitrary, but fixed secondary structure S^*, and the y-axis represents the Boltzmann probability $p(k)=\frac{Z_k}{Z}$ that a secondary structure has base pair distance k from S^*. Arguably, this probability distribution is an accurate one-dimensional projection of the rugged, high dimensional energy landscape near structure S^*, of the sort artistically rendered in the well-known energy landscape depicted in Figure 1 of [12]. In the sequel, we may call the FFTbor probability distribution a *structural neighbor profile*, or simply *structural profile* S^*. A hypothesis behind theoretical work in biomolecular folding theory in [13] is that kinetic folding

slows down as the energy landscape becomes more *rugged*. This is borne out in our computational experiments for RNA using FFTbor, as reported in Figure 2.

We randomly chose two TPP riboswitch aptamers from the seed alignment for Rfam family RF00059. The first sequence has EMBL accession code BX842649.1/277414-277318 and is comprised of the 97 nt sequence ACCUGACGCU AGGGGUGUUG GUGAAUUCAC CGACUGAGAA UAACCCUUUG AACCUGAUAG AGAUAAUGCU CGCGCAGGGA AGCAAGAAUA GAAAGAU, while the second sequence has EMBL accession code AACY022101973.1/389-487 and is comprised of the 99 nt sequence UAUAAGUCCA AGGGGUGCCA AUUGGCUGAG AUGGUUUUAA CCAAUCCCUU UGAACCUGAU CCGGUUAAUA CCGGCGUAGG AAUGGAUUUU CUCUACAGC. Rfam consensus and minimum free energy structures for both sequences are depicted in Figure 3. Despite the fact that there is no sequence homology according to pairwise BLAST [14], this figure clearly demonstrates that consensus and minimum free energy structures closely resemble each other, and that the structures of both TPP riboswitch aptamers are quite similar, with the exception of the leftmost hairpin loop [resp. multiloop]. The MFE structures differ from the consensus structures principally by the addition of base pairs not determined by covariation in the Rfam alignment. Indeed, if we let S_0, S_1 denote the Rfam consensus structure resp. MFE structure for the 97 nt sequence with EMBL accession code BX842649.1/277414-277318, then $S_0 \backslash S_1$ has 4 base pairs, and $S_1 \backslash S_0$ has 7 base pairs. If we let T_0, T_1 denote the Rfam consensus structure resp. MFE structure for the 99 nt sequence with EMBL accession code AACY022101973.1/389-487, then $T_0 \backslash T_1$ has 1 base pair, and $T_1 \backslash T_0$ has 5 base pairs.

We ran FFTbor on each of the TPP riboswitch aptamer sequences, with the MFE structure of each sequence taken as the initial structure S^* for that sequence. For the first sequence, BX842649.1/277414-277318, the FFTbor output suggests that there are low energy structures at a distance from the MFE structure, which might compete with the MFE structure and hence slow the kinetics of folding. In contrast, for the second sequence, AACY022101973.1/389-487, the FFTbor output suggests that there are no such competing low energy structures, hence the second sequence should fold more quickly than the first.

To test the hypothesis that folding is slower for rugged energy landscapes, we ran the kinetic folding software, Kinfold [15], on each of the two TPP riboswitch aptamer sequences, BX842649.1/277414-277318 and AACY022101973.1/389-487, to determine the *mean first passage time* (MFPT) to fold into the MFE structure, when starting from the empty structure. In this computational experiment, we took MFPT to be the average number of Monte Carlo steps taken by Kinfold, each step consisting of the addition or removal of a single base pair (or shift – see [15]), to fold the empty structure into the MFE structure, where the average was taken over 30 runs, with an absolute maximum number of Monte Carlo steps taken to be 500,000. The first sequence, BX842649.1/277414-277318, converged within 500,000 steps only for 20 out of 30 runs. Assigning the maximum step count of 500,000 for the 10 runs that did not converge, we found a mean first passage time of 311,075.06 steps for this sequence. The second sequence, AACY022101973.1/389-487, converged within 500,000 steps in 29 out of 30 runs, and we found a mean first passage time of 61,575.69 steps for this sequence. From computational experiments of this type, it is suggestive that FFTbor may prove useful in synthetic biology, where one would like to design rapidly folding RNA molecules that fold into a designated target structure. (See [16,17,18,19] for more on synthetic biology.) In particular, one could use RNAinverse [20], RNA-SSD [21], INFO-RNA [22], or our recent constraint programming exact solution of RNA inverse folding, RNAiFold (to appear in Journal of Bioinformatics and Computational Biology, see http://bioinformatics.bc.edu/clotelab/RNAiFold/), to output a list of sequences, whose minimum free energy structure is a designated target structure. Subsequently, using FFTbor, one could prioritize sequences in terms of FFTbor structural profile, on the grounds that sequences with a profile similar to the right panel of Figure 2 are likely to fold more rapidly than those whose profile resembles the left panel of Figure 2.

In order to more systematically determine the relation between kinetic folding speed and the ruggedness of an energy landscape near the MFE structure, we need to numerically quantify ruggedness. To this end, in the following we define the notion of *expected base pair distance* to a designated structure. Let S^* be an arbitrary secondary structure of the RNA sequence $\mathbf{s}=a_1,\ldots,a_n$. The expected base pair distance to S^* is defined by

$$E[\{d_{bp}(S,S^*) : S \in \mathbb{S}(a_1,\ldots,a_n)\}] = \sum_S P(S)\cdot d_{bp}(S,S^*) \quad (18)$$

where $\mathbb{S}(a_1,\ldots,a_n)$ denotes the set of secondary structures for $\mathbf{s}=a_1,\ldots,a_n$, $P(S)=\dfrac{\exp(-E(S)/RT)}{Z}$ is the Boltzmann probability of S, and $d_{bp}(S,S^*)$ denotes base pair distance between S and S^*. If we run FFTbor on an input sequence $\mathbf{s}$ and secondary structure S^*, then clearly $E[\{d_{bp}(S,S^*) : S \in \mathbb{S}(a_1,\ldots,a_n)\}] = \sum_k k\cdot p(k)$, where $p(k)=\dfrac{Z_k}{Z}$, obtained from the program output. If S^* is the empty structure, then FFTbor output is simply the probability distribution of the number of base pairs per secondary structure, taken over the Boltzmann ensemble of all structures.

For the benchmarking assay, we took all 61 selenocysteine insertion sequence (SECIS) sequences from the seed alignment of Rfam family RF00031 [23]. Average length was 64.32 ± 2.83 nt. For each sequence, we ran both FFTbor and a Monte Carlo folding algorithm, developed by E. Freyhult and P. Clote (unpublished). Using the Monte Carlo algorithm, we determined the mean first passage time (MFPT), defined as the average taken over 50 runs, of the number of Monte Carlo steps taken to fold the empty structure into the MFE structure, where an absolute upper bound of 5 million steps was allowed in the simulation. After unsuccessful attempts due to *ruggedness* of the energy landscape near the MFE structure, by using the Hartigan-Hartigan *dip* test of unimodality [24], expected base pair distance from MFE structure, total variation distance between FFTbor output and the exponential distribution estimated by the method of moments [25], etc., we ran FFTbor when starting from the empty structure (rather than the MFE structure) as initial structure. As mentioned above, in this case, FFTbor output is simply the probability distribution for the number of base pairs per structure, taken over the ensemble of all secondary structure for the input RNA sequence. Surprisingly, we found that there is a significant correlation of 0.48436192 with one-tailed p-value of 0.00018249 between the standard deviation of the FFTbor output (when starting from the empty structure) and logarithm base 10 of the mean first passage time. Table 1 and Figure 4 explain this phenomenon in detail.

In the right panel of Figure 4, we applied FFTbor to each of the two randomly chosen TPP riboswitch aptamers BX842649.1/277414-277318 and AACY022101973.1/389-487, starting from the empty reference structure $S^* = \varnothing$. The mean for the FFTbor

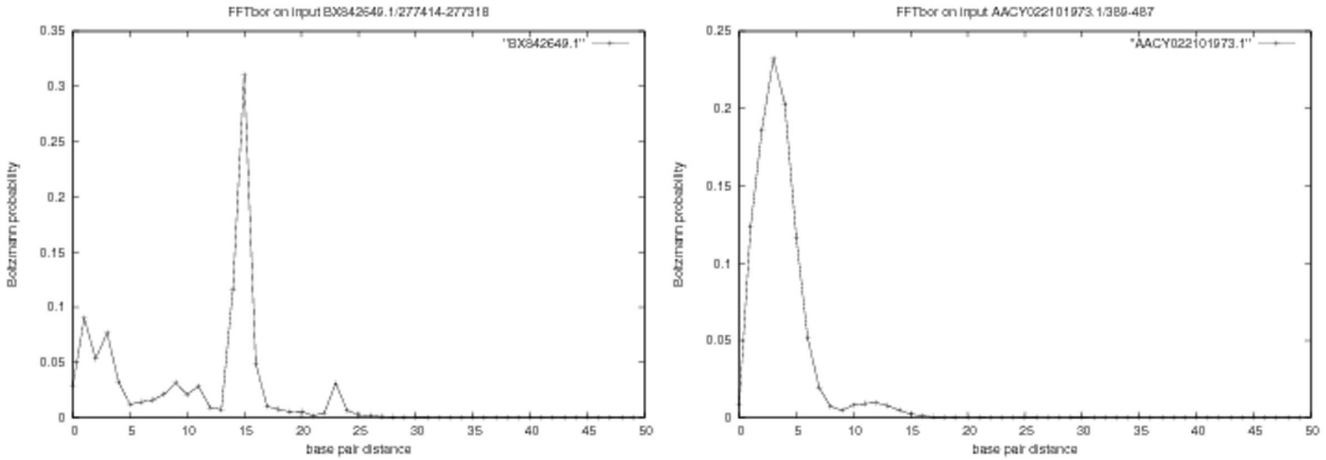

Figure 2. Output from FFTbor on two randomly selected thiamine pyrophosphate riboswitch (TPP) aptamers, taken from the Rfam database [23]. The x-axis represents base pair distance from the minimum free energy structure for each given sequence; the y-axis represents Boltzmann probabilities $p(k)=\frac{Z_k}{Z}$, where Z_k denotes the sum of Boltzmann factors or all secondary structures, whose base pair distance from the MFE structure is exactly k. *(Left)* The 97 nt sequence BX842649.1/277414-277318 appears to have a rugged energy landscape near its minimum free energy structure, with distinct low energy structures that may compete with the MFE structure during the folding process. *(Right)* The 99 nt sequence, AACY022101973.1/389-487 appears to have a smooth energy landscape near its MFE structure, with no distinct low energy structures to might compete with the MFE structure. Based on the FFTbor output or *structural profile* near MFE structure S^*, one might expect folding time for the first sequence to increase due to competition from metastable structures, while one might expect the second sequence to have rapid folding time. Computational Monte Carlo folding experiments bear out this fact. Kinfold [15] simulations clearly show that the second sequence folds at least four times more quickly than the first sequence. See text for details. Subfigure A Subfigure B Subfigure C Subfigure D.
doi:10.1371/journal.pone.0050506.g002

structural profile near the empty structure is $\mu_1=23.0203$ [resp. $\mu_2=27.5821$], the standard deviation σ for the FFTbor structural profile is $\sigma_1=2.22528791$ [resp. $\sigma_2=1.98565959$], and the Kinfold MFPT is 311,075.06 [resp. 61,575.69] for the TPP riboswitch aptamer AB030643.1/4176-4241 [resp. AL645723.11/192421-192359]. This anecdotal evidence supports the hypothesis that small standard deviation in FFTbor distribution is correlated with fast folding.

Additionally, in following a suggestion of one of the anonymous referees, we randomized the TPP riboswitches BX842649.1/277414-277318 and AACY022101973.1/389-487 by using our implementation of the Altschul-Erikson dinucleotide shuffle algorithm [26], and then applied FFTbor to these sequences, starting from the empty structure. The mean μ_1 and standard deviation σ_1 for the FFTbor distribution for randomized BX842649 are respectively $\mu_1=19.93$ and $\sigma_1=2.88$, while those for randomized AACY022101973 are $\mu_2=24.39$ and $\sigma_2=24.00$. Running Kinfold, with a maximum of 500,000 steps with 30 replicates (as explained in the text), we found that for randomized BX842649, all 30 runs converged yielding a mean first passage time (MFPT) of 13022.58 with standard deviation of 15221.78. In contrast for randomized AACY022101973, only 15 out of 30 runs converged within 500,000 steps, and discounting these nonconvergent data, we obtain an average mean first passage time (MFPT) of 94446.93 with standard deviation of 157107.43. This additional test provides more anecdotal evidence supporting our hypothesis that small standard deviation σ in FFTbor probability density is correlated with fast folding, as measured by MFPT.

Riboswitch expression platform prediction. A bacterial riboswitch is a portion of the 5′ untranslated region (UTR) of messenger RNA, that performs gene regulation by undergoing a conformational change upon binding with a ligand, such as

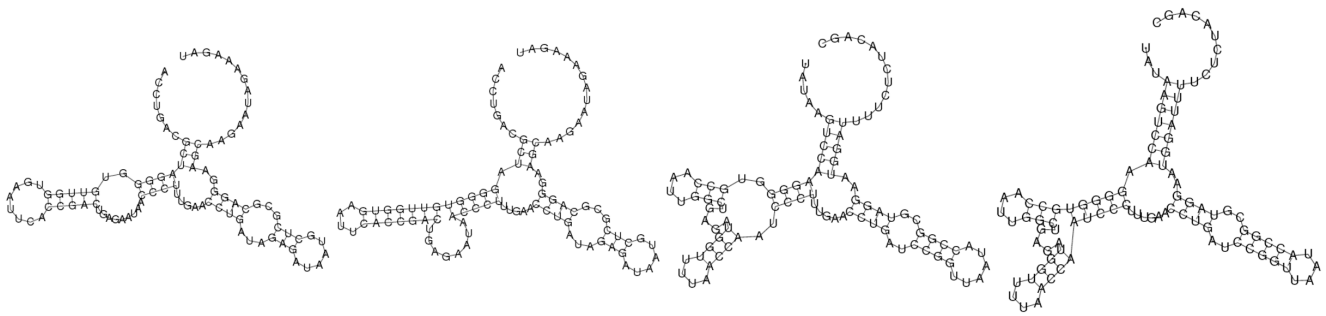

Figure 3. Rfam consensus structures (Rfam) and minimum free energy (MFE) secondary structures for two thiamine pyrophosphate (TPP) riboswitch aptamers, chosen at random from RF00059 Rfam family seed alignment [23]. Using pairwise BLAST [14], there is no sequence similarity, although the secondary structures are very similar, as shown in this figure. *(A)* Rfam consensus structure for BX842649.1/277414-277318. *(B)* MFE structure for BX842649.1/277414-277318. *(C)* Rfam consensus structure for AACY022101973.1/389-487. *(D)* Rfam consensus structure for AACY022101973.1/389-487.
doi:10.1371/journal.pone.0050506.g003

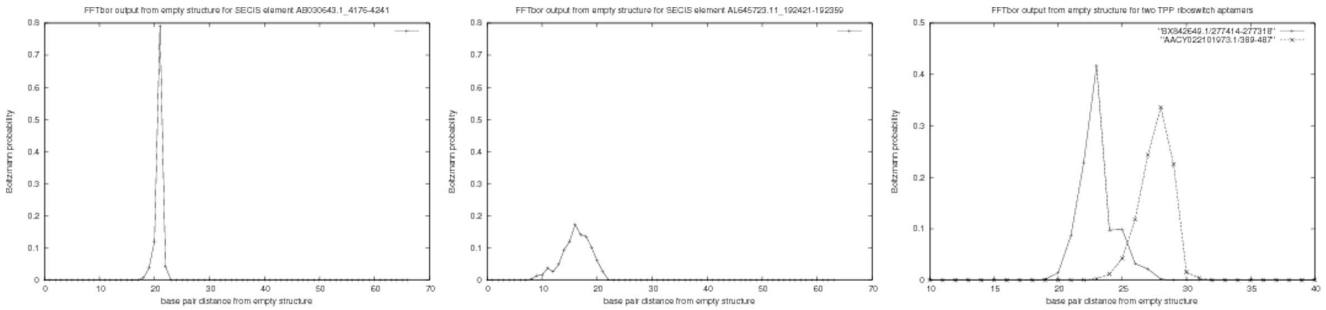

Figure 4. This figure represents the graphical output of FFTbor, when the empty structure is chosen as initial structure S^*. The x-axis represents the number of base pairs per structure, taken over the ensemble of all secondary structures for the given RNA sequence; the y-axis represents Boltzmann probability $p(k)=\frac{Z_k}{Z}$, where Z_k is the partition function for all secondary structures having exactly k base pairs. *(Left)* For the selenocysteine (SECIS) element AB030643.1/4176-4241 from Rfam family RF00031, the standard deviation σ of the number of base pairs, taken over the ensemble of all secondary structures, is 0.73, while the logarithm base 10 of the mean first passage time (logMFPT) is 4.75. *(Center)* For the selenocysteine (SECIS) element AL645723.11/192421-192359 from Rfam family RF00031, the standard deviation σ of the number of base pairs, taken over the ensemble of all secondary structures, is 2.68, while logMFPT is 5.69. Among the 61 sequences in the seed alignment of RF00031, AB030643.1/4176-4241 was the fastest folder, while AL645723.11/192421-192359 was the slowest folder. *(Right)* Superimposition of output of FFTbor for two TPP riboswitch aptamers: the 97 nt sequence BX842649.1/277414-277318 and the 99 nt sequence AACY022101973.1/389-487, both obtained when taking the empty structure for the initial structure S^*. The mean μ for the FFTbor structural profile near the empty structure is 23.02 [resp. 27.5821], the standard deviation σ for the FFTbor structural profile is 2.23 [resp. 1.99], and the Kinfold MFPT is $311{,}075.06$ [resp. $61{,}575.69$] for the TPP riboswitch aptamer AB030643.1/4176-4241 [resp. AL645723.11/192421-192359]. The right panel of this figure should be compared with Figure 2. These anecdotal results bear up the correlation between standard deviation σ and logMFPT described in Table 1.
doi:10.1371/journal.pone.0050506.g004

guanine, thiamine pyrophosphate, lysine, etc. [27]. This conformational change may either turn on or off the corresponding gene by either transcriptional or translational regulation of the messenger RNA [28], depending on the particular riboswitch. The common feature shared by all riboswitches is that a gene is regulated by conformational change upon ligand binding. Bacterial riboswitches are often found upstream of operons, regulating groups of genes, as in purine *de novo* synthesis and salvage [29].

A riboswitch consists of two equally important parts: an upstream *aptamer*, capable of highly discriminative binding to a particular ligand, and a downstream *expression platform*, capable of undergoing a radical conformational change upon binding of a ligand with the discriminating aptamer. Since aptamers have been

Table 1. Pearson correlation between various aspects of selenocysteine insertion sequences from the seed alignment of Rfam family RF00031 [23].

	μ	σ	$\frac{\sigma}{\mu}$	len	MFE	logMFPT
μ	1					
σ	−0.43722448	1				
$\frac{\sigma}{\mu}$	−0.691411183	0.943650913	1			
len	0.707683898	−0.158951202	−0.364591789	1		
MFE	−0.569474125	0.739515083	0.759622716	−0.368485646	1	
logMFPT	−0.036291124	0.48436192	0.376230235	0.405865529	0.399015556	1

For each of the 61 RNA sequences, we ran FFTbor, starting from empty initial structure S^*, and we ran a Monte Carlo folding algorithm, developed by E. Freyhult and P. Clote (unpublished). Using the Monte Carlo algorithm, we determined the mean first passage time (MFPT), defined as the average taken over 50 runs, of the number of Monte Carlo steps taken to fold the empty structure into the MFE structure, where an absolute upper bound of 5 million steps was allowed in the simulation. From the output of FFTbor, we computed *(1)* the mean number (μ) of base pairs per structure, taken over the ensemble of all secondary structures for the given sequence, *(2)* the standard deviation (σ) of the number of base pairs per structure, *(3)* the coefficient of variation $\frac{\sigma}{\mu}$, *(4)* the RNA sequence length, and *(5)* the minimum free energy (MFE). Additionally, we computed the logarithm base 10 of mean first passage time (log10MFPT), taken over 50 Monte Carlo runs per sequence (log base 10 of the standard deviation of number of Monte Carlo steps per run was approximately 9% of log10MFPT on average). The table shows the correlation between each of these aspects. Some correlations are obvious – for example, *(i)* the standard deviation σ is highly correlated with the coefficient of variation $\frac{\sigma}{\mu}$; *(ii)* the mean μ is negatively correlated with the coefficient of variation $\frac{\sigma}{\mu}$; *(iii)* the mean μ is negatively correlated with the minimum free energy (MFE) – if most low energy structures in the ensemble have many base pairs, then it is likely that the minimum free energy is very low (i.e. since MFE is negative, the absolute value of MFE increases); *(iv)* sequence length is negatively correlated with MFE – as sequence length increases, the minimum free energy (MFE) decreases. However, it may appear surprising that *(v)* the mean μ number of base pairs per structure is independent of MFPT (correlation -0.036291124), although *(vi)* MFE is correlated with MFPT (correlation 0.399015556) – i.e. from *(iii)*, lower MFE is correlated with a larger average μ number of base pairs per structure, from *(vi)* higher MFE is correlated with longer folding time, but from *(v)* the average μ number of base pairs per structure is independent of folding time. The most important insight from this table is that *(vii)* standard deviation σ is correlated with mean first passage time – the correlation is statistically significant, with one-tailed p-value of 0.00018249.
doi:10.1371/journal.pone.0050506.t001

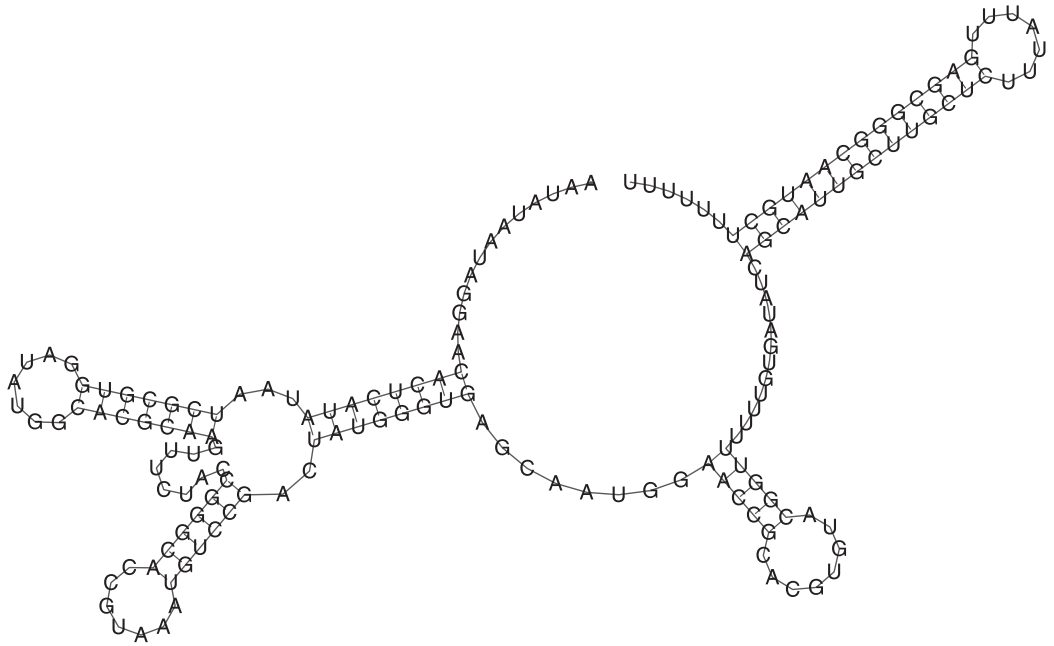

Figure 5. Gene OFF secondary structure of the xpt G-box purine riboswitch in *B. subtilis*; structure taken from that in Figure 1A of [27].
doi:10.1371/journal.pone.0050506.g005

under strong evolutionary pressure to bind with high affinity (e.g. $K_D \approx 5$ nM for guanine [30]), there is strong sequence conservation found in the aptameric region of orthologous riboswitches. In contrast, while secondary structure is conserved in the terminator loop of the expression platform in purine riboswitches, there is relatively low sequence conservation (data not shown). While a number of methods exist to computationally predict riboswitch aptamers [31,32,33,34,35] (and especially INFERNAL [36], which latter is used to predict riboswitch aptamers in Rfam), it is an important biological problem to determine the expression platform, since the structure of the expression platform can suggest whether there is transcriptional regulation via a terminator loop or translational regulation via the sequestration of the Shine-Dalgarno sequence [28]. Determination of the precise location and structure of the expression platform is difficult due to low conserved sequence identity (in-house computations, data not shown). Although this problem remains open, we report here how FFTbor may provide help to biologists in the selection and prioritization of riboswitch candidates.

Figure 5 depicts the gene OFF structure of the xpt G-box purine riboswitch in *B. subtilis*, as determined by inline-probing – this structure was taken from Figure 1 of [27]. Note that this structure

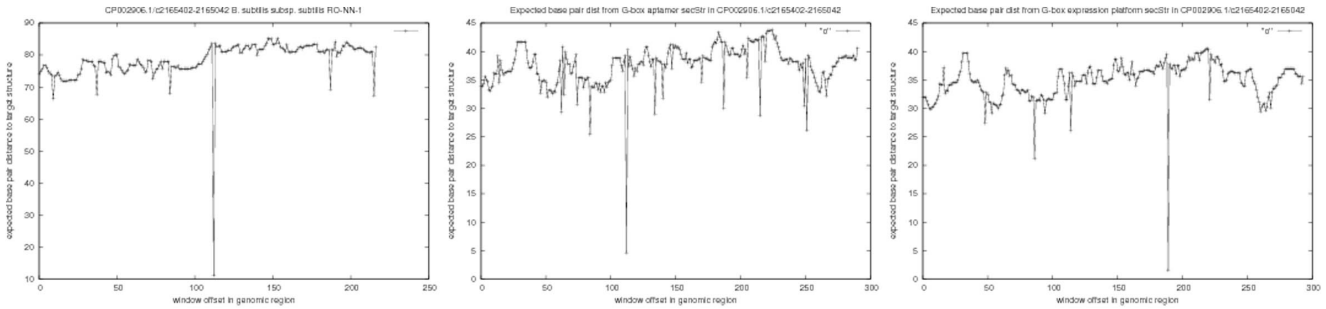

Figure 6. Graph of the expected distance from target secondary structure, as a function of window offset position in the 5′ untranslated region (UTR) of the xpt gene of *B. subtilis*; i.e. GenBank accession code CP002906.1/c2165402-2165042 *B. subtilis* subsp. subtilis RO-NN-1. In a moving window application, FFTbor computed the Boltzmann probability $p(k)$ that secondary structures of the current window contents have base pair distance k from the target (or initial) structure S^*. In each case, the size of the window was set to equal the length of S^*. *(Left)* Target structure S^* comprises the entire secondary of the xpt riboswitch, as depicted in Figure 5, with the exception that the leading and trailing unpaired positions were removed, as explained in the text – see displayed dot bracket structure in (19). *(Center)* Target structure S^* comprises only the aptamer secondary structure, as displayed in dot bracket structure in (20). *(Right)* Target structure S^* comprises only the expression platform secondary structure, as displayed in dot bracket structure in (21). The number of points displayed on the x-axis differs in each case, since the window size differs, as explained above. The very well-defined minimum in each panel corresponds to the exact location of the entire riboswitch (left panel), aptamer (center panel) and expression platform (right panel). Note that the base line value for the expected base pair distance in the left panel (entire riboswitch) is approximately 70, while that for both the center panel (aptamer) and right panel (expression platform) is approximately 35.
doi:10.1371/journal.pone.0050506.g006

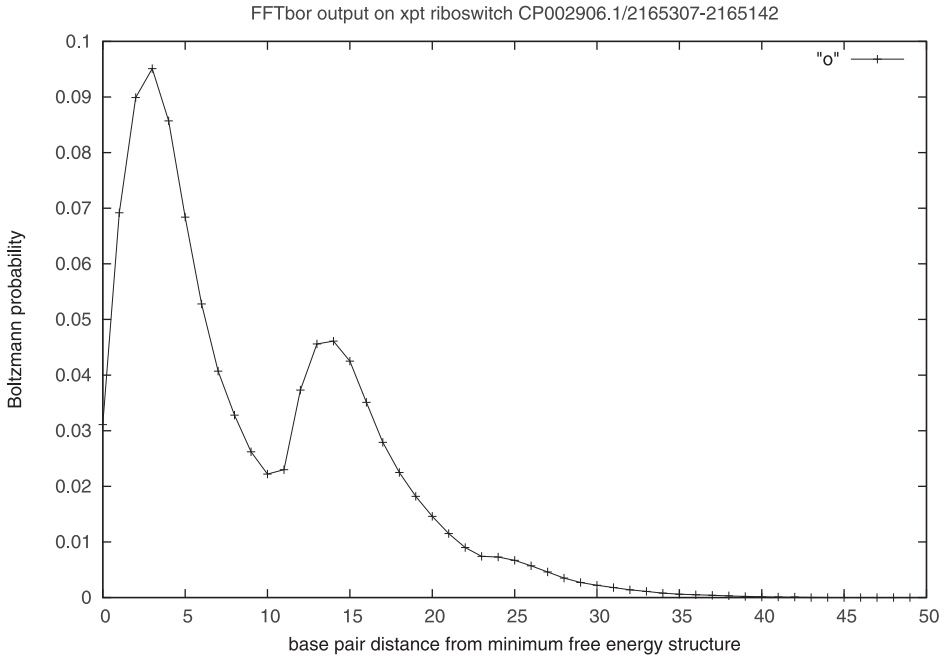

Figure 7. The 161 nt xpt G-box purine riboswitch described in Figure 1A of [27], found on the complement strand of GenBank accession number CP002906.1/c2165302-2165142 in complete genome of *B. subtilis subsp. subtilis RO-NN-1*. We extended this 161 nt sequence to a sequence of length 200 nt, by appending flanking downstream genomic nucleotides. The web site http://bioinformatics.bc.edu/clotelab/FFTbor displays a movie of all prefixes of the resulting 200 nt sequence, where prefix lengths range from 70,72,74,...,200.
doi:10.1371/journal.pone.0050506.g007

is only partial, since there are regions with no base pairs depicted, despite the fact that additional base pairs could be added. By using blastn, it is found that this 161 nt purine riboswitch can be found on the complement strand of GenBank accession number CP002906.1/c2165302-2165142 in complete genome of *B. subtilis subsp. subtilis RO-NN-1*. Figure 6 depicts the result of three computational experiments with FFTbor. The left panel displays the expected base pair distance to the following secondary structure

$$\begin{array}{l}.(((((((...(((((((.....)))))))).....((((((.....))))))..)))))))..... \\ (((((.....))))).....(((((((((((((.....)))))))))))))).\end{array} \quad (19)$$

as a function of window offset, where window size equals the size of this target structure. This structure was obtained by removing all leading and trailing unpaired positions from the structure depicted in Figure 5, except for the leftmost [resp. rightmost] unpaired position adjacent to the leftmost [resp. rightmost] base-paired position. The reason for removal of the leading and trailing unpaired positions was that the structure of [27], depicted in Figure 5, is clearly only partial, as earlier mentioned. The center panel displays the expected base pair distance to the following secondary structure

$$.(((((((...(((((((.....)))))))).....((((((.....))))))..))))))). \quad (20)$$

as a function of window offset, where window size equals the size of this target *aptamer* structure. Similarly, the right panel displays the expected base pair distance to the following secondary structure

$$.(((((.....))))).....(((((((((((((.....))))))))))))). \quad (21)$$

as a function of window offset, where window size equals the size of this target *expression platform* structure. Figure 6 determines the precise location of the xpt riboswitch, both aptamer and expression platform.

If the biologically functional target structure is unknown, one can instead attempt a similar moving window computation, where the target structure is taken to be the minimum free energy structure of the current window contents. In this case, one may hope to determine a bimodal distribution, as displayed in Figure 7. Given an input RNA sequence, or genomic region, the web server http://bioinformatics.bc.edu/clotelab/FFTbor creates a movie as follows, described here for the xpt riboswitch previously discussed. We extended the 161 nt xpt G-box purine riboswitch described in Figure 5, with GenBank accession number CP002906.1/c2165302-2165142, to a sequence of length 200 nt, by appending flanking downstream genomic nucleotides. Running FFTbor on all prefixes of the resulting sequence of lengths 70,72,74,...,200, we produced a movie, displayed on the webserver http://bioinformatics.bc.edu/clotelab/FFTbor. Figure 5 displays the output of FFTbor on the 166 nt prefix, clearly showing a bimodal distribution. Attempting to automate the identification of non-unimodal FFTbor output, we have applied the Hartigan-Hartigan dip-test [24], implemented in R; however, the dip-test appears to be too sensitive, in that a probability distribution is reported to be non-unimodal, even when visual inspection indicates that it appears overwhelmingly to be unimodal (data not shown). It is for this reason that the web server http://bioinformatics.bc.edu/clotelab/FFTbor produces a movie of prefixes, where the user can start/stop the movie, move forward/backward, or download all raw data output by FFTbor.

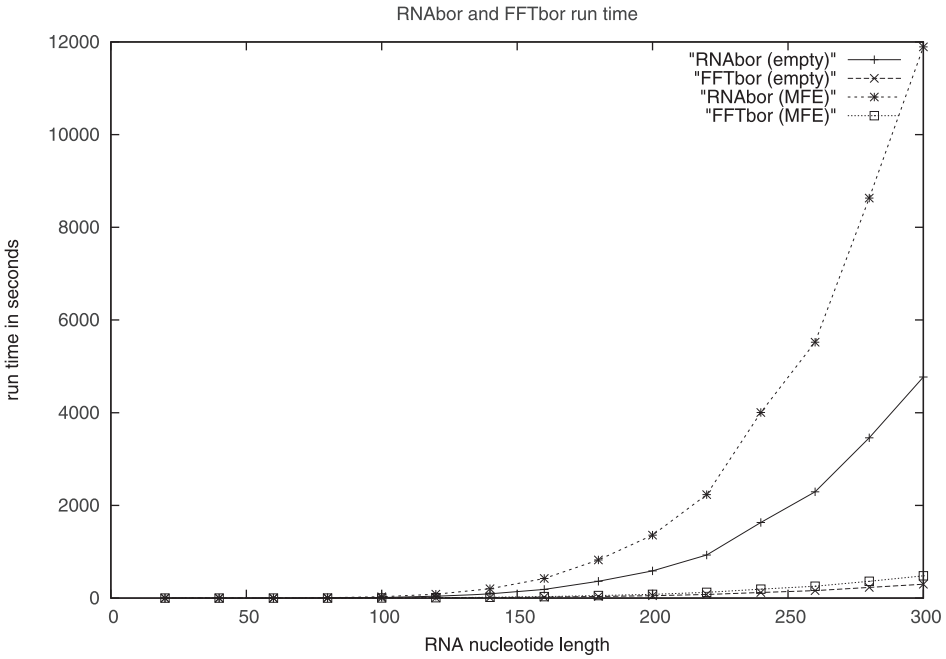

Figure 8. Run times in seconds for RNAbor and FFTbor, on random RNA of length $20,40,60,\ldots,300$ in step size of 20 nt. Each algorithm was run with the empty initial structure S^*, see rows RNAbor (empty), FFTbor (empty), and with the minimum free energy structure as the initial structure S^*, see rows RNAbor (MFE) and FFTbor (MFE). Note that for both RNAbor and FFTbor, the run time increases when S^* is the MFE structure, rather than the empty structure. Notice the radical improvement in the run time of FFTbor over that of RNAbor.
doi:10.1371/journal.pone.0050506.g008

Benchmarking results

Total variation distance for density of states. Recall that the *total variation distance* between two probability distributions $P=\{p_x : x\in\Omega\}$ and $Q=\{q_x : x\in\Omega\}$, defined on the same sample space Ω, is defined by

$$\delta(P,Q)=\frac{\sum_{x\in\Omega}|p_x-q_x|}{2}.$$

The *density of states* for an RNA sequence s with respect to an initial structure S^* of s is defined to be the probability distribution $P=(p_0,\ldots,p_{n-1})$ where $p_k=\mathbf{Z}^k_{1,n}/\mathbf{Z}$. In all our tests, for RNA of length up to 400 nt, we found the total variation distance between

Len	1	2
200	123.2 ± 16.2	61.8 ± 8.0
250	331.1 ± 27.2	166.1 ± 13.7
300	723.4 ± 59.9	365.2 ± 30.1
350	1,380.8 ± 95.2	698.4 ± 46.9
400	2,239.1 ± 210.9	1,129.5 ± 104.3
450	3,635.0 ± 857.4	1,980.9 ± 126.5
500	5,076.7 ± 1,292.1	3,389.8 ± 788.4

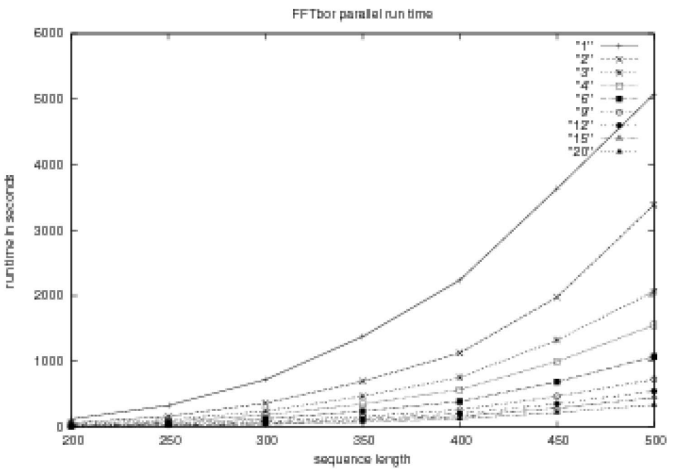

Figure 9. *(Left)* Table showing parallel run times in seconds for FFTbor, using OpenMP http://openmp.org/. Column headers 1,2 indicate the number of cores used in the computational experiment. For each sequence length $200,\ldots,500$, five random RNAs were generated using equal probability for each nucleotide A,C,G,U. Run time in seconds, plus or minus one standard deviation, are given for a 24-core AMD Opteron 6172 with 2.10 GHz and 64 GB RAM, with only 1 (resp. 2) cores used. *(Right)* Graph showing parallel run time of FFTbor on an AMD Opteron 6172 with 2.10 GHz and 64 GB RAM, using respectively 1,2,3,4,6,9,12,15,20 cores.
doi:10.1371/journal.pone.0050506.g009

Table 2. Table showing parallel run times of FFTbor, using OpenMP http://openmp.org/. Column headers 2,3, etc. indicate the number of cores used in the computational experiment.

Len	2	3	4	6	9	12	15	20
200	61.8±8.0	41.6±6.0	31.6±4.2	21.1±2.9	15.0±2.3	11.3±1.1	9.6±1.3	7.6±1.5
250	166.1±13.7	111.5±8.7	84.1±7.1	56.9±4.0	38.8±3.8	30.3±2.6	24.6±1.6	18.9±2.7
300	365.2±30.1	246.4±20.5	184.5±14.9	125.1±9.6	85.5±6.6	64.9±6.5	53.6±5.9	42.1±5.2
350	698.4±46.9	470.1±32.6	352.0±23.0	242.6±15.8	163.3±12.4	125.1±6.7	104.2±8.9	76.2±3.9
400	1,129.5±104.3	757.5±68.9	571.6±53.9	391.2±36.4	265.1±24.2	207.2±18.4	165.6±14.5	125.9±14.9
450	1,980.9±126.5	1,326.3±85.1	1,000.0±59.0	688.9±44.8	469.2±29.1	355.1±25.4	289.8±21.2	223.1±18.2
500	3,389.8±788.4	2,067.9±99.2	1,555.0±72.2	1,074.3±53.7	728.1±40.9	548.3±24.0	451.5±25.7	338.1±22.7

For each sequence length 200, . . . ,500, five random RNAs were generated using equal probability for each nucleotide A,C,G,U. Run time in seconds, plus or minus one standard deviation, are given for a 24-core AMD Opteron 6172 with 2.10 GHz and 64 GB RAM. Least-squares fit of the data indicates a quadratic dependency of run time on sequence length (despite the obvious $O(n^4)$ theoretical run time), and a power law dependence of approximately $x^{-0.99}$ on the number of cores x.
doi:10.1371/journal.pone.0050506.t002

P, as computed to 6 decimal places by RNAbor and by FFTbor, to be 0. It follows that FFTbor can reliably be used in place of RNAbor to determine Boltzmann probabilities $p(k) = P(d_{BP}(S,S^*)=k)$.

Run time comparison of RNAbor and FFTbor. As visible from the defining recursions, the algorithmic time complexity of RNAbor is $O(n^5)$ and space complexity is $O(n^3)$, where n is the length of input RNA sequence. In contrast, the time complexity of FFTbor is $O(n^4)$ and space complexity is $O(n^2)$. Figure 8 displays run time curves for both RNAbor and FFTbor, when the initial structure S^* is taken to be either the empty structure or the minimum free energy (MFE) structure.

Here, we compare the run time of RNAbor [1] and the (unparallelized version of) FFTbor, using a Dell Power Edge 1950, 2× Intel Xeon E5430 Quad core with 2.80 GHz and 16 GB RAM. For $n=20,40,60,\ldots,300$, in step size of 20 nt, we generated n random RNA sequences of length n with equal probability for each nucleotide A,C,G,U (i.e. a 0th order Markov chain). For values of $n\le 200$, 100 random sequences of length n were generated, while for values of $220\le n\le 300$, only 10 sequences of length n were generated. RNA sequences larger than 300 nt were not tested, due to $O(n^3)$ memory constraints required by RNAbor. For each RNA sequence, RNAbor and FFTbor were both run, each starting with empty initial structure S^*, and also with initial sequence S^* taken to be the MFE structure. Each data point in the table comprises the average run time for three independent evaluations.

OpenMP parallelization of FFTbor. OpenMP is a simple and flexible multi-platform shared-memory parallel programming environment, that supports parallelizations of C/C++ code – see http://openmp.org/. Using OpenMP primitives, we created multiple threads to evaluate the polynomial $\mathcal{Z}(x)$ on different complex nth roots of unity. The table in the left panel of Figure 9 and Table 2 together present benchmarks, executed on a 24-core AMD Opteron 6172 with 2.10 GHz and 64 GB RAM, for the speedup of FFTbor as a function of the number of cores. The table in Figure 9 describes average run time in seconds (± one standard deviation) for running FFTbor on random RNA of length 200,250,300,400,450,500 with either 1 or 2 cores. Table 2 presents similar data for running FFTbor on 2,3,6,4,12,15,20 cores. Although FFTbor clearly has quartic $O(n^4)$ run time as a function of RNA sequence length, least-squares fit of run times from Table 2 instead shows a quadratic run time for RNA sequences of length up to 500 nt. There appears to be a *power law* dependence of FFTbor speedup, as a function of number of cores. For instance, for random RNA of length 200 nt, least-squares fit of the data from the table yields a run time of $105.29x^{-0.923}$ with R^2 value of 0.99782. A power law behavior is demonstrated, with similarly high R^2 values, for each fixed sequence length in Table 2, with different coefficients of variable x but with approximately the same exponent of x (data not shown, but easily computable from data in Table 2).

Conclusion and Discussion

In this paper, we have used a dynamic programming computation to evaluate the polynomial

$$\mathcal{Z}(x)=\sum_{k=0}^{n} c_k x^k \tag{22}$$

on the complex nth roots of unity $1,e^{2\pi i/n},\ldots,e^{2\pi i(n-1)/n}$, where the coefficients $c_k=\mathbf{Z}_{1,n}^k$ are equal to the sum of Boltzmann factors over all secondary structures of a given RNA sequence, whose base pair distance to a given initial structure S^* is k. Recall the definition of polynomial

$$p(x)=\frac{\mathcal{Z}(x)}{\mathbf{Z}_{1,n}}=\sum_{k=0}^{n} p_k x^k \tag{23}$$

obtained from $\mathcal{Z}(x)$, whose coefficients are Boltzmann probabilities $p_k=\dfrac{\mathbf{Z}_{1,n}^k}{\mathbf{Z}_{1,n}}$ that a secondary structure has base pair distance k to S^*. By using the fast Fourier transform to compute the inverse discrete transform, we can approximate to m decimal places the coefficients $p_k=\dfrac{\mathbf{Z}_{1,n}^k}{\mathbf{Z}_{1,n}}=\dfrac{c_k}{\mathbf{Z}_{1,n}}$ of $p(x)$, and thus the m most significant positions of $c_k=\mathbf{Z}_{1,n}^k=p_k\cdot\mathbf{Z}_{1,n}$. Interpolation is performed for $p(x)$, rather than $\mathcal{Z}(x)$, due to issues concerning numerical instability. The computational advantage of FFTbor over its predecessor RNAbor [1] is that the new algorithm runs in quartic time $O(n^4)$ and quadratic space $O(n^2)$, in contrast to the $O(n^5)$ run time and $O(n^3)$ space required by RNAbor. We have additionally provided a parallelization of FFTbor using OpenMP primitives. Additionally, we have described applications of FFTbor to determine the correlation between kinetic folding speed and the

ruggedness of the energy landscape, and to predict the location of riboswitch expression platform candidates.

It is important to point out that the algorithm and software RNAbor is more general than that of FFTbor – in particular, RNAbor not only computes the partition function values $\mathbf{Z}_{1,n}^{k}$, for all $0 \leq k \leq n$, but as well as computes the structures S_k, defined to be the minimum free energy structure over all k-neighbors of initial structure S^*. In contrast, FFTbor only computes the m most significant digits of the probabilities $p_k = \mathbf{Z}_{1,n}^{k}/\mathbf{Z}$, for $0 \leq k \leq n$, where by multiplication of p_k by the partition function $Z = \mathbf{Z}_{1,n}$, one obtains an approximation of the partition function values $\mathbf{Z}_{1,n}^{k}$. There is no possibility that FFTbor can compute the structures S_k, nor can at present we see how to use FFTbor to sample structures from the Boltzmann ensemble of structures having base pair distance k from S^*.

In [37,38], we introduced the a related *parametric* RNA structure algorithm, RNAmutants, which computes the partition function $\mathbf{Z}_{1,n}^{k}$ and minimum free energy structure $MFE(k)$ over all secondary structures of all k-point mutants of a given RNA sequence $\mathbf{s} = s_1, \ldots, s_n$. In [39], RNAmutants was extended to sample low energy structures over k-point mutants within a certain range of GC-content. Some of the ideas in [39] foreshadowed the results of this paper, and in the future, we intent to apply interpolation and the FFT to similarly provide a more efficient version of RNAmutants. Nevertheless, this future, more efficient version will be incapable of efficiently sampling low energy structures over k-point mutants, analogous to the current differences between RNAbor and FFTbor.

Supporting Information

File S1 Supplementary information.
(PDF)

Acknowledgments

FFTbor depends heavily on the use of the Fast Fourier Transform implementation FFTW of Frigo and Johnson [40] at http://www.fftw.org/.

Author Contributions

Conceived and designed the experiments: PC YP. Performed the experiments: PC ES SS. Analyzed the data: PC ES. Contributed reagents/materials/analysis tools: PC ID ES SS. Wrote the paper: PC. Interpolation idea: YP. Idea for handling numerical instability (FFT, scaling): PC. Designed the software used in analysis: PC ES SS.

References

1. Freyhult E, Moulton V, Clote P (2007) Boltzmann probability of RNA structural neighbors and riboswitch detection. Bioinformatics 23: 2054–2062.
2. Clote P, Lou F, Lorenz W (2012) Maximum expected accuracy structural neighbors of an RNA secondary structure. BMC Bioinformatics 13: S6.
3. Ding Y, Lawrence CE (2003) A statistical sampling algorithm for RNA secondary structure prediction. Nucleic Acids Res 31: 7280–7301.
4. Dotu I, Lorenz WA, VAN Hentenryck P, Clote P (2010) Computing folding pathways between RNA secondary structures. Nucleic Acids Res 38: 1711–1722.
5. Flamm C, Hofacker IL, Maurer-Stroh S, Stadler PF, Zehl M (2001) Design of multistable RNA molecules. RNA 7: 254–265.
6. Matthews D, Sabina J, Zuker M, Turner D (1999) Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. J Mol Biol 288: 911–940.
7. Xia T, J SantaLucia J, Burkard M, Kierzek R, Schroeder S, et al. (1999) Thermodynamic parameters for an expanded nearest-neighbor model for formation of RNA duplexes with Watson-Crick base pairs. Biochemistry 37: 14719–35.
8. Freyhult E, Moulton V, Gardner P (2005) Predicting RNA structure using mutual information. Appl Bioinformatics 4: 53–59.
9. Nussinov R, Jacobson AB (1980) Fast algorithm for predicting the secondary structure of single stranded RNA. Proceedings of the National Academy of Sciences, USA 77: 6309–6313.
10. Higham N (2004) The numerical stability of barycentric Lagrange interpolation. IMA J Numer Anal 24: 547–556.
11. Cormen T, Leiserson C, Rivest R (1990) Algorithms. McGraw-Hill. 1028 pages.
12. Wolynes PG (2005) Energy landscapes and solved protein-folding problems. Philos Transact A Math Phys Eng Sci 363: 453–464.
13. Bryngelson JD, Onuchic JN, Socci ND, Wolynes PG (1995) Funnels, pathways, and the energy landscape of protein folding: a synthesis. Proteins 21: 167–195.
14. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ (1990) Basic local alignment search tool. J Mol Biol 215: 403–410.
15. Flamm C, Fontana W, Hofacker I, Schuster P (2000) RNA folding at elementary step resolution. RNA 6: 325–338.
16. Shetty RP, Endy D, Knight TF (2008) Engineering BioBrick vectors from BioBrick parts. J Biol Eng 2: 5.
17. Knight TF (2005) Engineering novel life. Mol Syst Biol 1: 2005–2020.
18. Waldminghaus T, Kortmann J, Gesing S, Narberhaus F (2008) Generation of synthetic RNA-based thermosensors. Biol Chem 389: 1319–1326.
19. Zadeh JN, Wolfe BR, Pierce NA (2011) Nucleic acid sequence design via efficient ensemble defect optimization. J Comput Chem 32: 439–452.
20. Hofacker I (2003) Vienna RNA secondary structure server. Nucleic Acids Res 31: 3429–3431.
21. Andronescu M, Fejes A, Hutter F, Hoos H, Condon A (2004) A new algorithm for rna secondary structure design. J Mol Biol 336: 607–624.
22. Busch A, Backofen R (2006) Info-rna, a fast approach to inverse rna folding. Bioinformatics 22: 1823–1831.
23. Gardner PP, Daub J, Tate J, Moore BL, Osuch IH, et al. (2011) Rfam: Wikipedia, clans and the "decimal" release. Nucleic Acids Res 39: D141–D145.
24. Hartigan J, Hartigan P (1985) The dip test of unimodality. Ann Statist 13: 70–84.
25. Zar J (1999) Biostatistical Analysis. Prentice-Hall, Inc.
26. Altschul S, Erikson B (1985) Significance of nucleotide sequence alignments: A method for random sequence permutation that preserves dinucleotide and codon usage. Mol Biol Evol 2(6): 526–538.
27. Serganov A, Yuan Y, Pikovskaya O, Polonskaia A, Malinina L, et al. (2004) Structural basis for discriminative regulation of gene expression by adenine- and guanine-sensing mRNAs. Chem Biol 11(12): 1729–1741.
28. Tucker BJ, Breaker RR (2005) Riboswitches as versatile gene control elements. Curr Opin Struct Biol 15: 342–348.
29. Mandal M, Boese B, Barrick J, Winkler W, Breaker R (2003) Riboswitches control fundamental biochemical pathways in Bacillus subtilis and other bacteria. Cell 113(5): 577–586.
30. Mandal M, Breaker RR (2004) Adenine riboswitches and gene activation by disruption of a transcription terminator. Nat Struct Mol Biol 11: 29–35.
31. Bengert P, Dandekar T (2004) Riboswitch finder–a tool for identification of riboswitch RNAs. Nucl Acids Res 32: W154–159.
32. Chang TH, Huang HD, Wu LC, Yeh CT, Liu BJ, et al. (2009) Computational identification of riboswitches based on RNA conserved functional sequences and conformations. RNA 15: 1426–1430.
33. Abreu-Goodger C, Merino E (2005) RibEx: a web server for locating riboswitches and other conserved bacterial regulatory elements. Nucleic acids research 33.
34. Singh P, Bandyopadhyay P, Bhattacharya S, Krishnamachari A, Sengupta S (2009) Riboswitch Detection Using Profile Hidden Markov Models. BMC Bioinformatics 10: 325+.
35. Bergig O, Barash D, Nudler E, Kedem K (2004) STR2: a structure to string approach for locating G-box riboswitch shapes in pre-selected genes. In Silico Biol 4: 593–604.
36. Nawrocki EP, Kolbe DL, Eddy SR (2009) Infernal 1.0: inference of RNA alignments. Bioinformatics.
37. Clote P, Waldispühl J, Behzadi B, Steyaert JM (2005) Exploring the energy landscape of k-point mutagens of rna. Bioinformatics 21: 4140–4147.
38. Waldispuhl J, Devadas S, Berger B, Clote P (2008) Efficient algorithms for probing the RNA mutation landscape. PLoS Comput Biol 4: e1000124.
39. Waldispühl J, Ponty Y (2011) An unbiased adaptive sampling algorithm for the exploration of RNA mutational landscapes under evolutionary pressure. Journal of Computational Biology 18: 1465–79.
40. Frigo M, Johnson SG (2005) The design and implementation of FFTW3. Proceedings of the IEEE 93: 216–231.